Offloading Media Traffic to P4 Programmable Data Plane Switches

Elie Kfoury[1], Jorge Crichigno[1], Elias Bou-Harb[2], Vladimir Gurevich[3]

[1]University of South Carolina, Columbia, SC
[2]University of Texas, San Antonio, TX
[3]Barefoot Networks, an Intel Company

CI Lab @ UofSC: http://ce.sc.edu/cyberinfra

CI Engineering Brown Bag
Friday, February 28, 2020

AGENDA

- Introduction
- Background Information
  - Session Initiation Protocol (SIP) and Real Time Protocol (RTP)
  - Network Address Translation (NAT) traversal problem
  - P4 switches
- Proposed solution
- Evaluation
- Lessons learned

INTRODUCTION

- According to estimations, media traffic represents approximately 80% of the total traffic over the Internet[1]
- Much media traffic is generated by end users communicating with each other
- Media services (voice, video) running alongside the data network in campuses are becoming standard

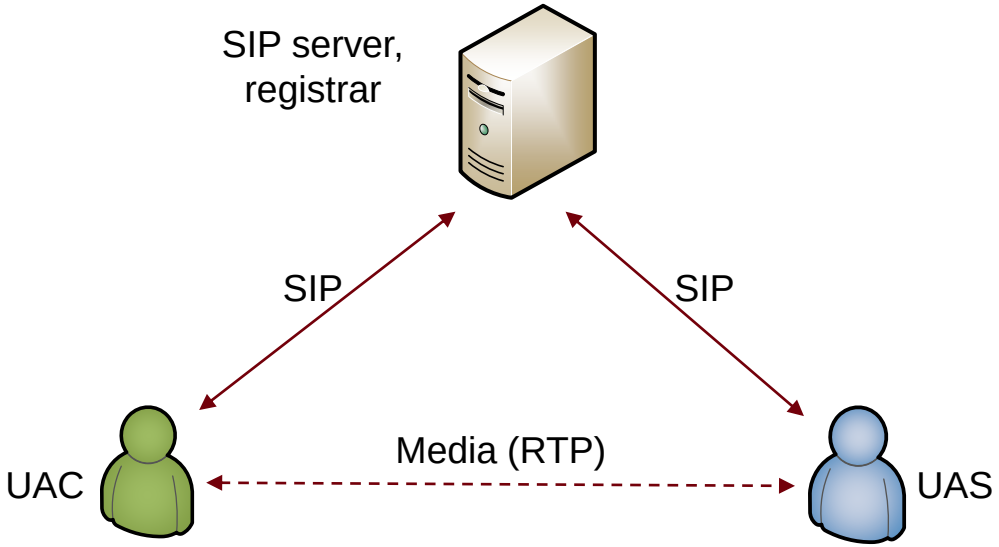
1. H. W. Barz and G. A. Bassett, Multimedia networks: protocols, design and applications, John Wiley and Sons, 2016.

VOICE AND VIDEO

- Conversational Voice- and Video-over-IP are widely used today
  - Open and proprietary (Skype, WhatsApp) solutions
- Supporting protocols are divided into two main categories
  - Session control protocols (signaling)
    - ✓ Session Initiation Protocol (SIP)
    - ✓ Establish and manage the session
  - Media protocols (media)
    - ✓ Real Time Protocol (RTP)
    - ✓ Transfer audio and video streams between the end-users
- Desirable Quality-of-Service (QoS) characteristics
  - Delay- and jitter-sensitive, low values
  - Occasional losses are tolerated

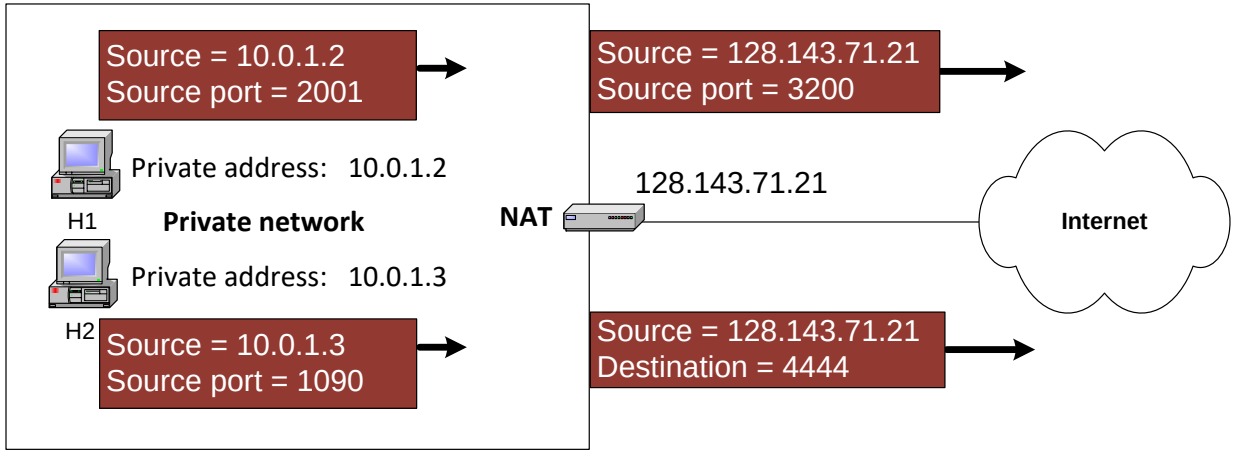
SIGNALING AND MEDIA PROTOCOLS

- SIP initiates, maintains, and terminates multimedia sessions between endpoints
  - User agent client (UAC)
  - User agent server (UAC)
- RTP transports real-time data, such as audio and video



NETWORK ADDRESS TRANSLATION

- Network Address Translation (NAT)
  - Maps ports, private IP addresses to public IP addresses
- Used in campus / enterprise networks, operators[1]
- NAT introduces various issues
  - Violation of the end-to-end principle
  - Traversal of end-to-end sessions



[1]I. Livadariu et al., “Inferring carrier-grade NAT deployment in the wild,” in IEEE 2018 INFOCOM, 2018.

NETWORK ADDRESS TRANSLATION

- NAT prevents a user from outside from initiating a session
- If both users have NATs, then neither can accept a call
  - IP translation is recorded by a SIP registrar server
- SIP carries the IP addresses and ports to be used by RTP to send/receive media
  - **NAT-translated IP, ports are unknown until RTP traffic starts**
- Several solutions proposed for NAT traversal
  - STUN - RFC 5389[1], TURN - RFC 7566[2], ICE - RFC 8445[3]

1. D. Wing, P. Matthews, R. Mahy, and J. Rosenberg, “RFC 5389 - STUN: Session traversal utilities for NAT,” 2008.
2. M. Petit-Huguenin, S. Nandakumar, G. Salgueiro, and P. Jones, “RFC 7566 - TURN: Traversal using relays around NAT (TURN) uniform resource identifiers,” 2013.
3. J. Rosenberg and C. Holmberg, “RFC 8445 - ICE: Interactive connectivity establishment: a protocol for Network Address Translator (NAT) traversal,” 2018.

RELAY SERVER

- Intermediary device

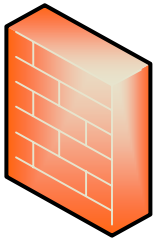
SIP server

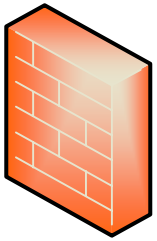
Relay server

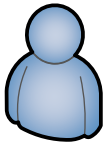
RTP Information at relay server

	Device IP - port	Allocated IP - port
A		
B		

A



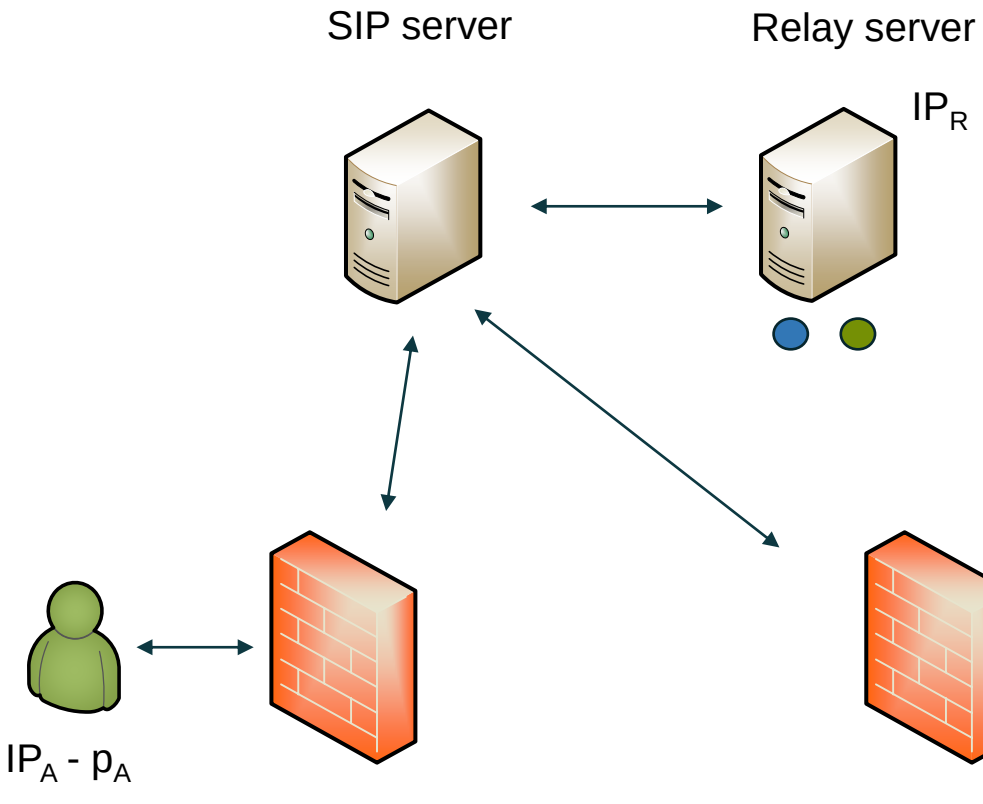




B

RELAY SERVER

- Intermediary device
- SIP establishes the session
  - RTP ports are unknown
  - The relay server allocates one port for each device

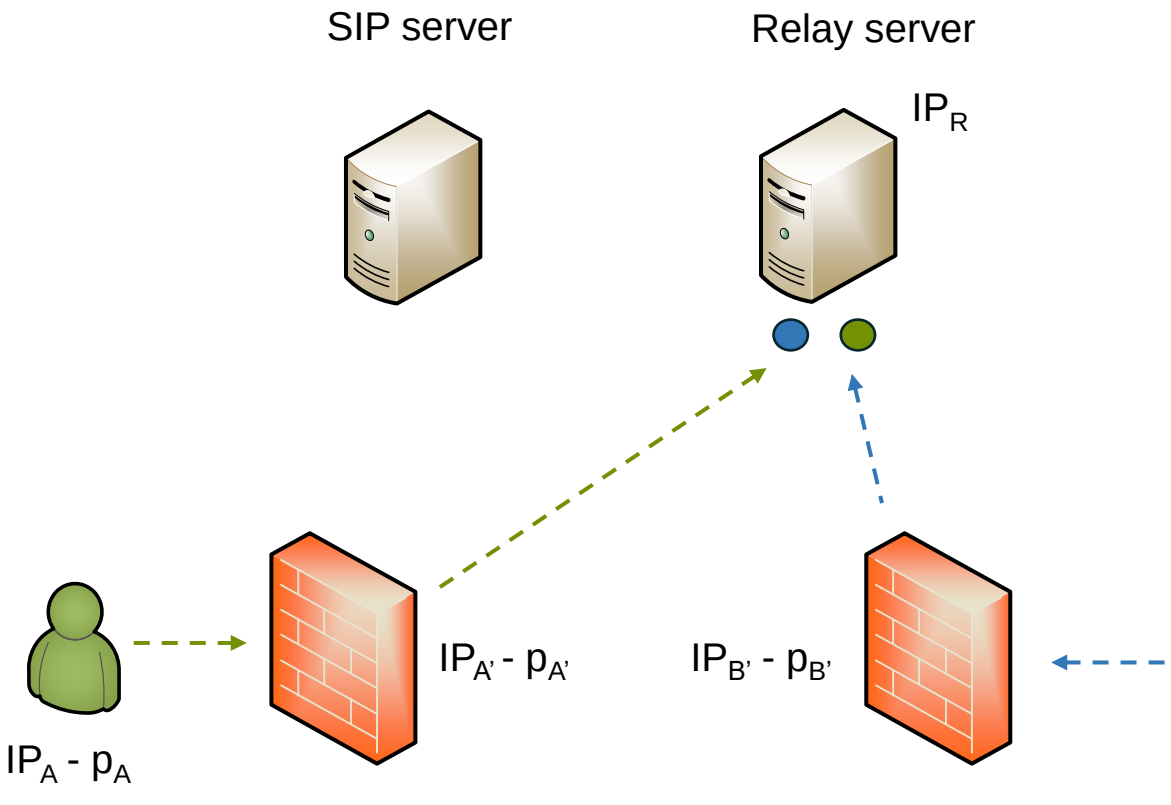


RTP Information at relay server

	Device IP - port	Allocated IP - port
A	-- --	IP_R - $p_{R\text{-}A}$
B	-- --	IP_R - $p_{R\text{-}B}$

RELAY SERVER

- Intermediary device
- SIP establishes the session
  - RTP ports are unknown
  - The relay server allocates one port for each device
- The relay server receives and relays the RTP traffic

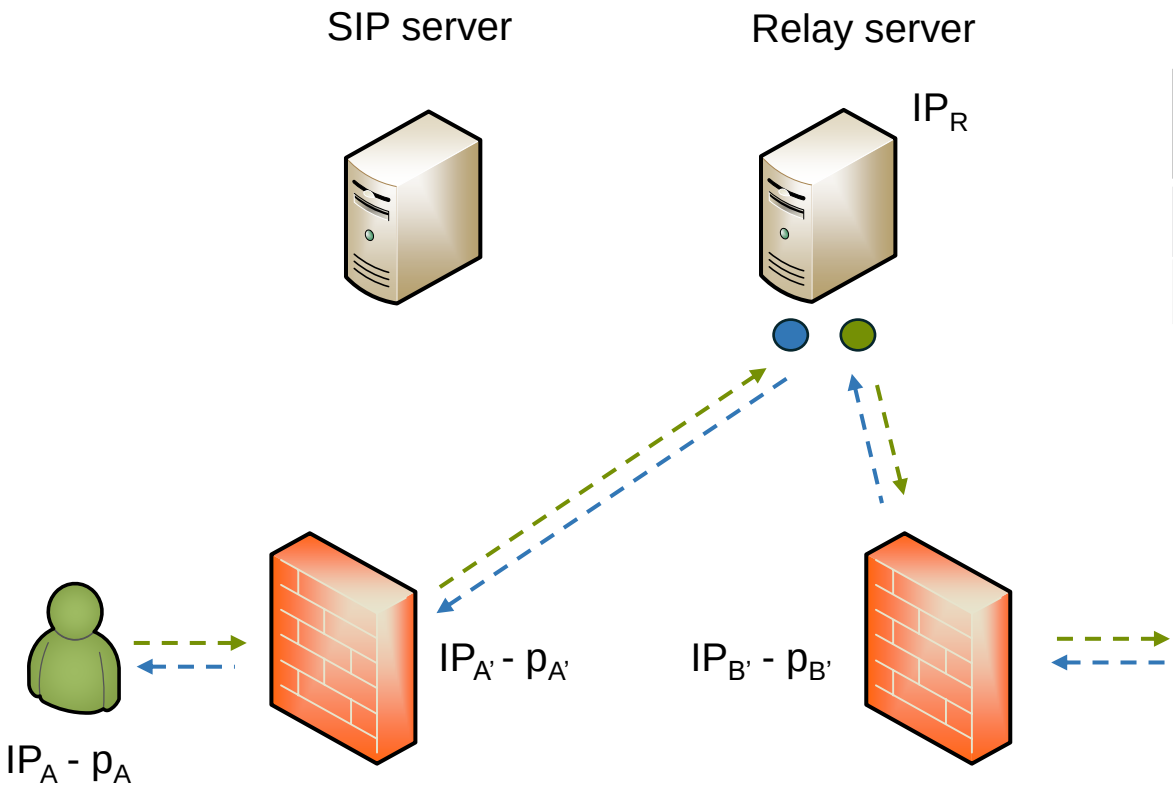


RTP Information at relay server

	Device IP - port	Allocated IP - port
A	$IP_{A'} - p_{A'}$	$IP_R - p_{R\text{-}A}$
B	$IP_{B'} - p_{B'}$	$IP_R - p_{R\text{-}B}$

RELAY SERVER

- Intermediary device
- SIP establishes the session
  - RTP ports are unknown
  - The relay server allocates one port for each device
- The relay server receives and relays the RTP traffic



RTP Information at relay server

	Device IP - port	Allocated IP - port
A	$IP_{A'} - p_{A'}$	$IP_R - p_{R\text{-}A}$
B	$IP_{B'} - p_{B'}$	$IP_R - p_{R\text{-}B}$

OVERVIEW P4 SWITCHES

- P4 switches permit programmer to program the data plane
- Add proprietary features
  - Parse packet headers, including UDP
  - Header inspection; identify media session using the 5-tuple
  - Modify fields; IP addresses and ports

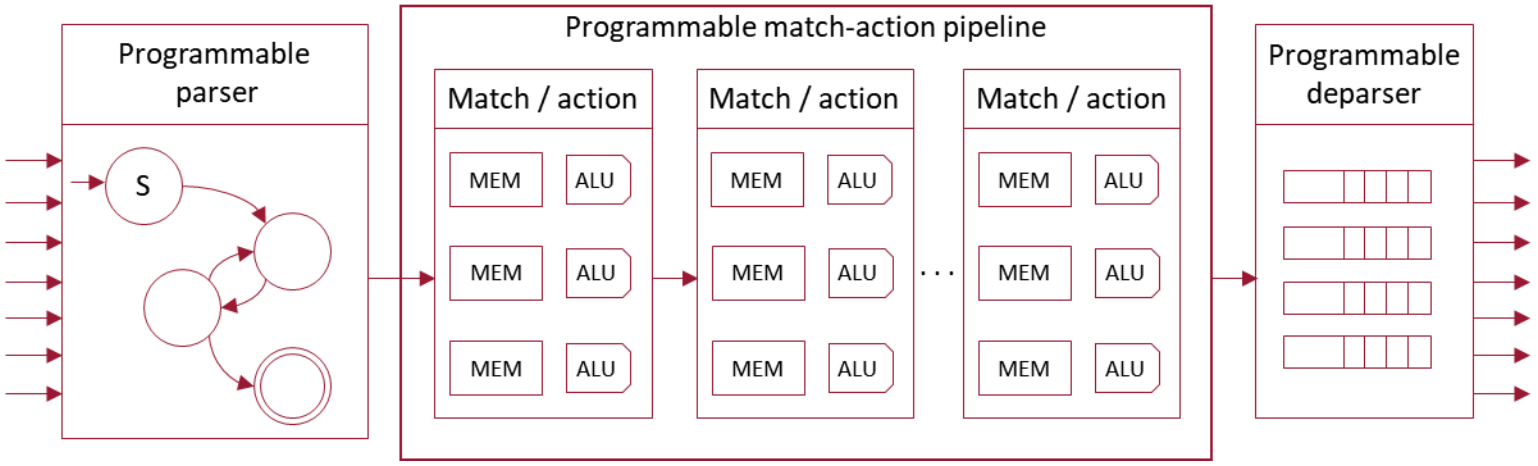
```
136  /*************************************************************************
137  ************************ P A R S E R  ***********************************
138  *************************************************************************/
139
140      state parse_ethernet {
141          packet.extract(hdr.ethernet);
142          transition select(hdr.ethernet.etherType) {
143              TYPE_IPV4: parse_ipv4;
144              default: accept;
145          }
146      }
147
148      state parse_ipv4 {
149          packet.extract(hdr.ipv4);
150          verify(hdr.ipv4.ihl >= 5, error.IPHeaderTooShort);
151          transition select(hdr.ipv4.ihl) {
152              5             : accept;
153              default       : parse_ipv4_option;
154          }
155      }
```

P4 code

Programmable chip

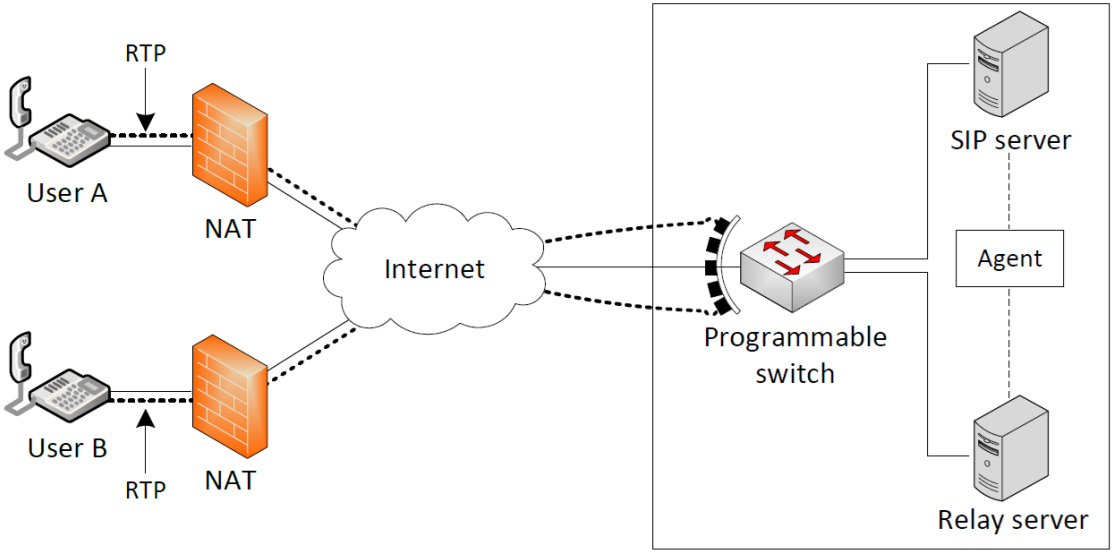
PISA ARCHITECTURE

- Several programmable switches implement the Protocol Independent Switch Architecture (PISA)
  - Abstract processing model
  - Programmers specify how a packet should be parsed and processed through match-action tables
- If the P4 program compiles, it runs on the chip at line rate



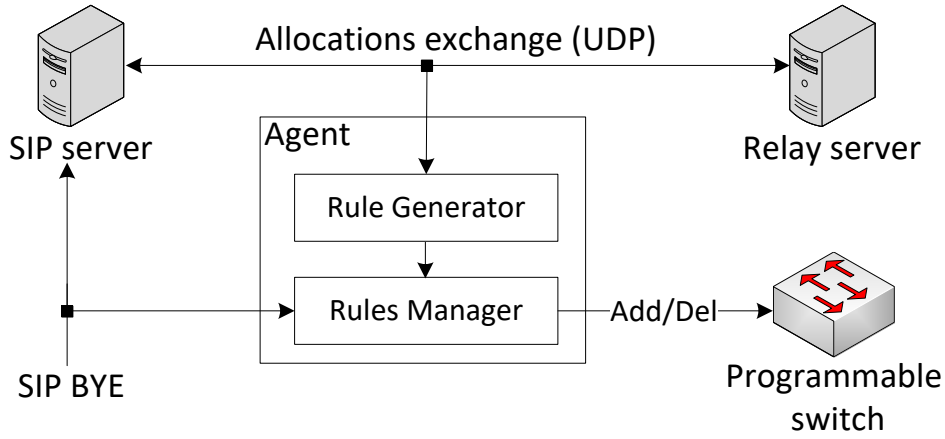
PROPOSED SYSTEM

- The proposed architecture uses programmable switches to emulate the behavior of the relay server:
  1. Parse the incoming packet carrying media traffic from the first party, say user A
  2. Identify the session this packet belongs to by using the 5-tuple
  3. Replace the source IP with that of the relay server, and the source port with that used by the relay server to receive traffic from user A
  4. Replace the destination IP and the destination port with those of user B
  5. Recalculate both IPv4 and UDP checksums
  6. Forward the packet to user B



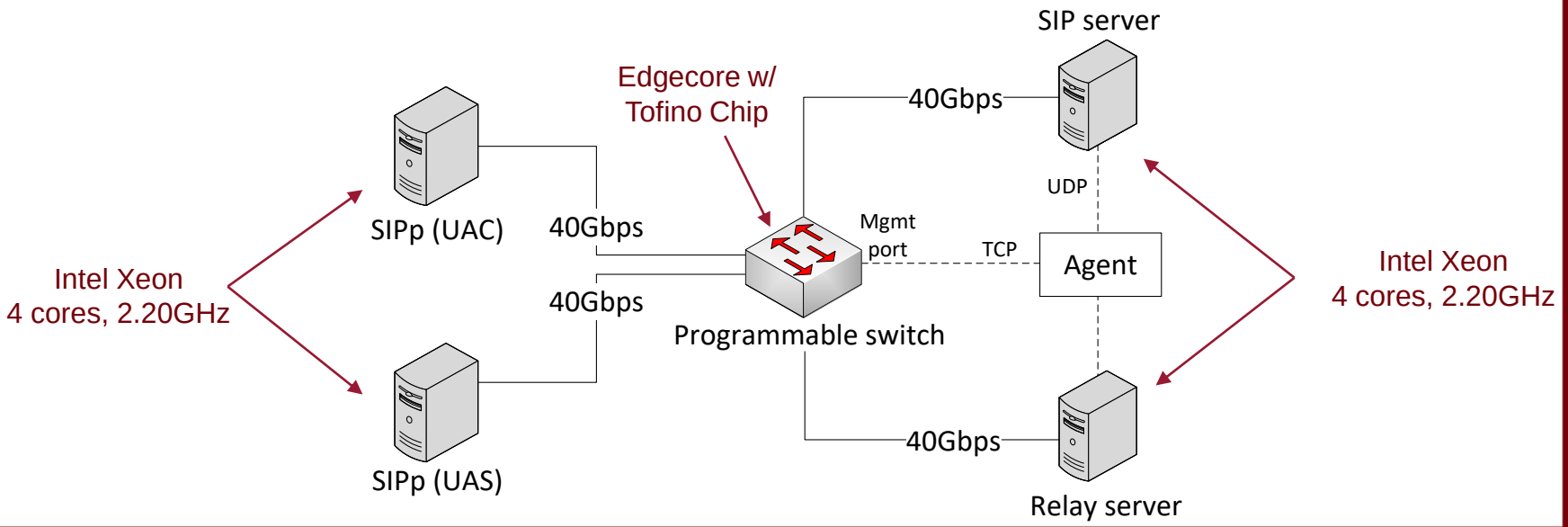
PROPOSED SYSTEM (CONT'D)

- A custom software (agent) learns the ports allocated to a media session by the relay server
- The Rule Generator uses the 5-tuple allocated to the media session to construct a unique session identifier
- It stores identifiers of the media sessions and the new headers' values in the switch
- It also clears media sessions allocated in the switch when a call is teared down



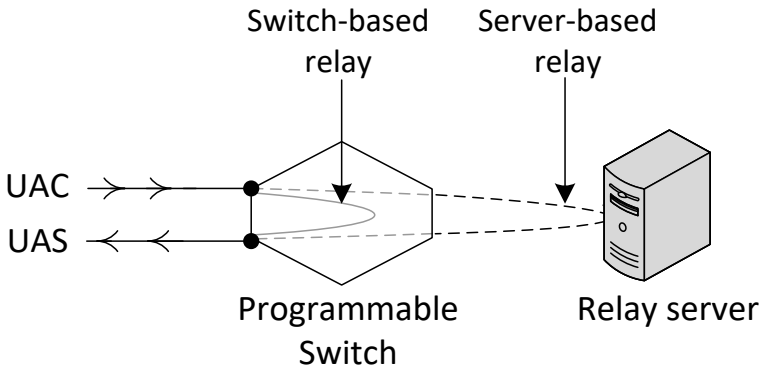
IMPLEMENTATION AND EVALUATION

- System components
  - OpenSIPS, an open source implementation of a SIP server
  - RTPProxy, a high-performance relay server for RTP streams
  - SIPp: an open source SIP traffic generator that can establish multiple concurrent sessions and generate media (RTP) traffic
  - Iperf3: traffic generator used to generate background UDP traffic
  - Edgecore Wedge100BF-32X: programmable switch



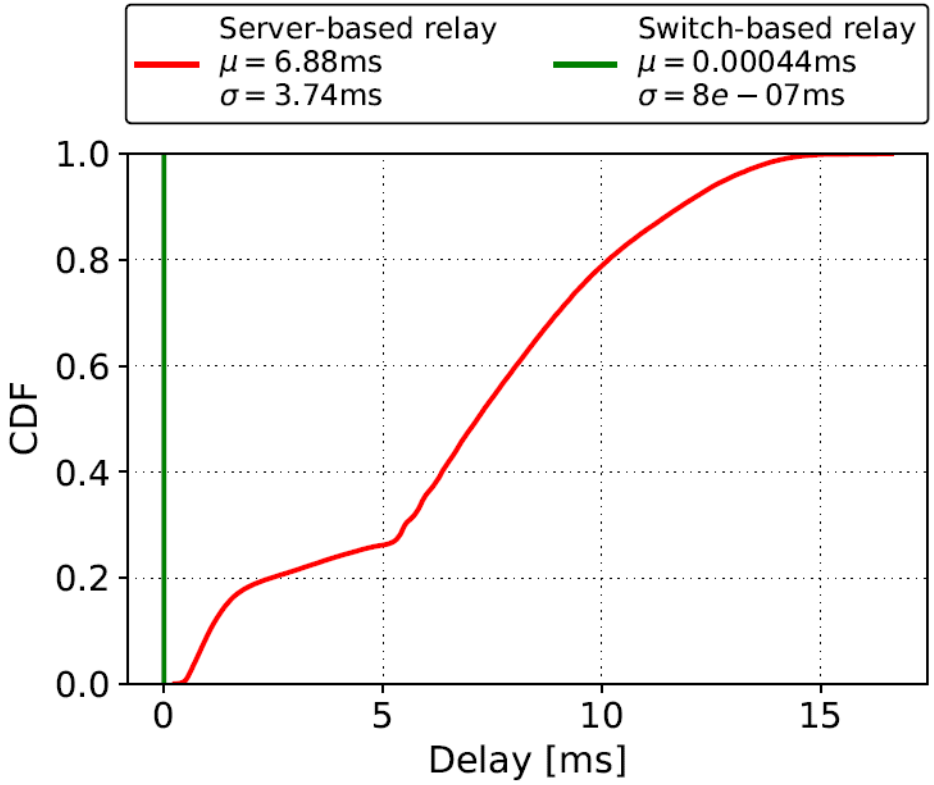
IMPLEMENTATION AND EVALUATION

- Two scenarios are considered:
  1. "Server-based relay": relay server is used to relay media between end devices, without the intervention of the switch
  2. "Switch-based relay": the switch is used to relay media
- UAC (SIPp) generates 900 media sessions
- The rate at which sessions arrive is 30 per second
- The test lasts for 300 seconds
- G.711 media encoding codec (160 bytes every 20ms)



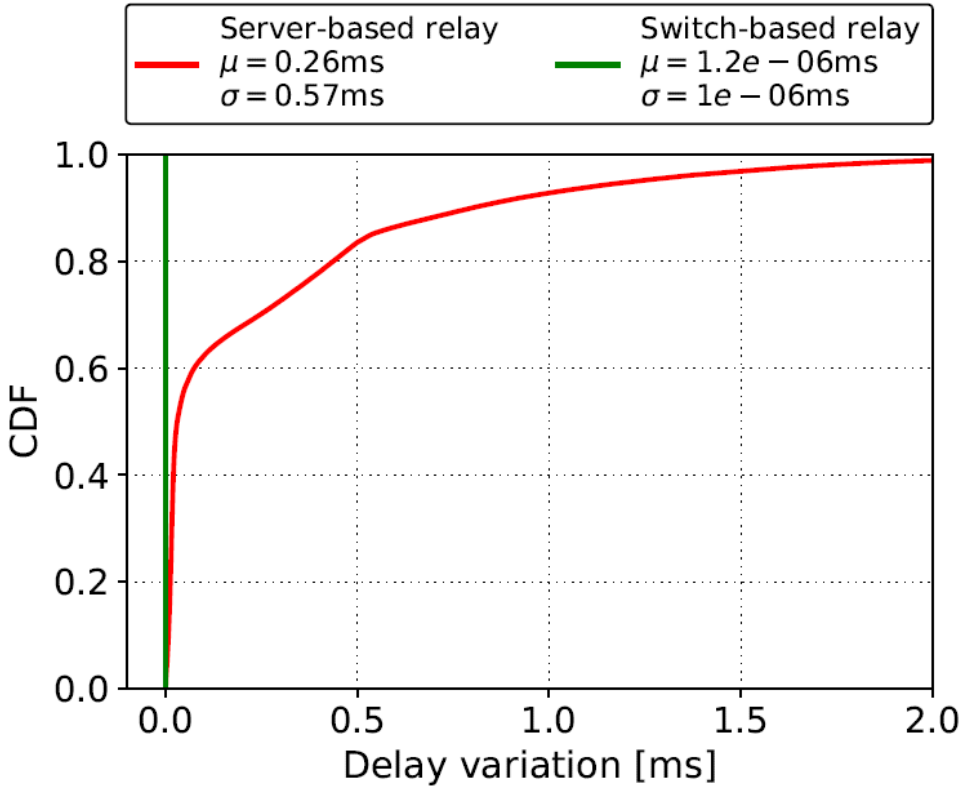
RESULTS

- **Delay**: the time interval starting when a packet is received from the UAC by the switch's ingress port and ending when the packet is forwarded by the switch's egress port to the UAS
  - Delay contributions of the switch and the relay server



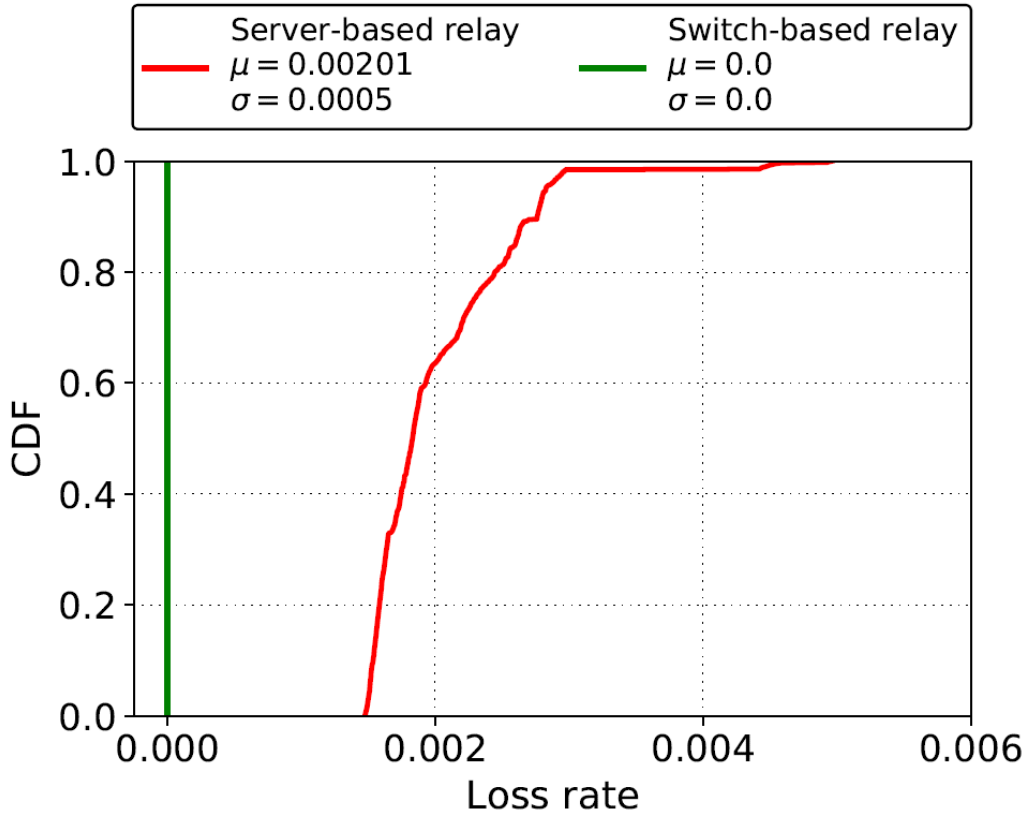
RESULTS

- **Delay variation**: the absolute value of the difference between the delay of two consecutive packets
  - Analogous to jitter, as defined by RFC 4689



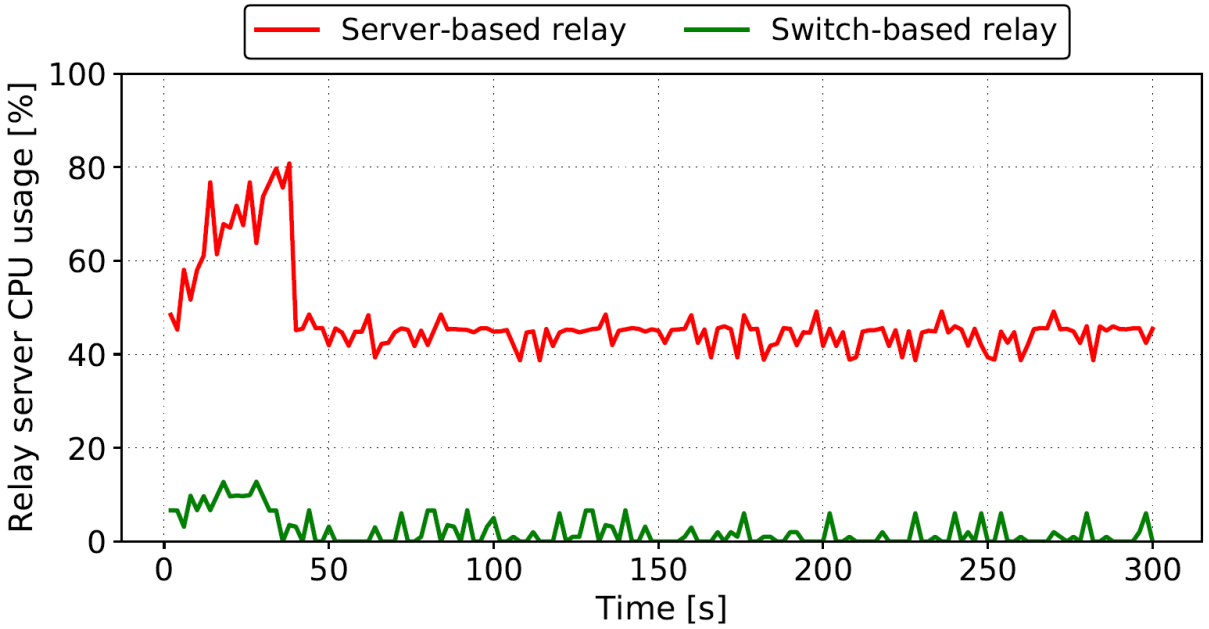
RESULTS

- **Loss rate**: number of packets that fail to reach the destination
  - Calculation is based on the sequence number of the RTP header



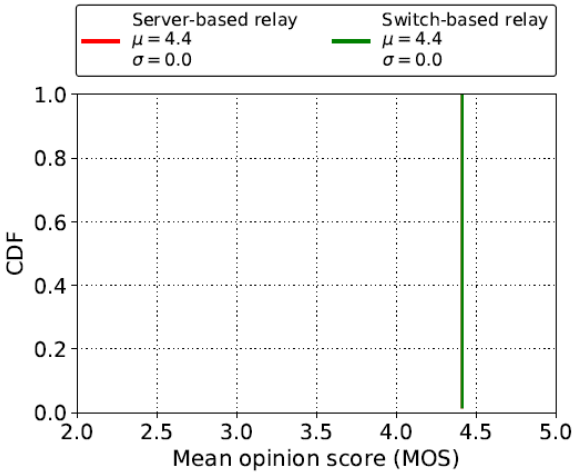
RESULTS

- **CPU usage**: the percentage of the CPU's capacity used by the relay server

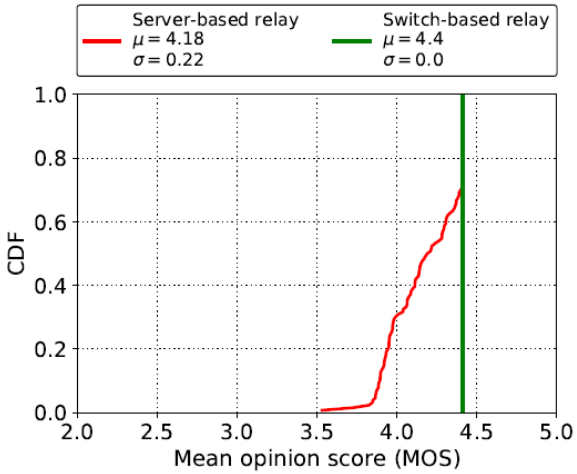


RESULTS

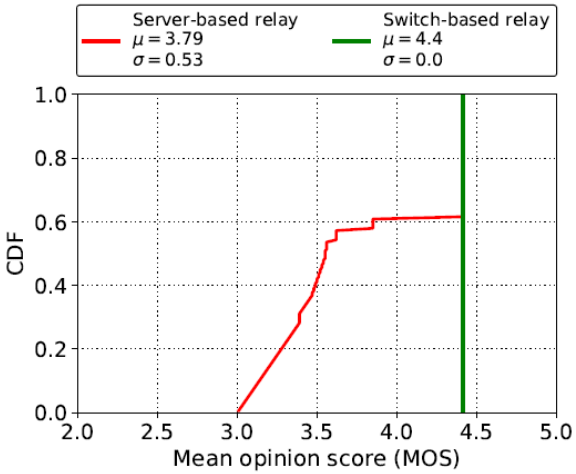
- **Mean Opinion Score (MOS):** estimation of the quality of the media session
  - A reference quality indicator standardized by ITU-T
  - Maximum for G.711 is ~4.4



(a) 750 simultaneous sessions.



(b) 1500 simultaneous sessions.



(c) 1800 simultaneous sessions.

RESOURCE CONSUMPTION

- The prototype is implemented in two different scenarios:
  - On top of the baseline switch program (switch.p4)
    - ✓ Implements various features including Layer 2/3 functionalities, ACL, QoS, etc.
  - Standalone implementation

On top of switch.p4			
Table size	**SRAM**	**Hash Bits**	**TCAM**
32,000	+8.45%	+2.7%	+0%
64,000	+16.2%	+4.6%	+0%
Standalone program			
Table size	**SRAM**	**Hash Bits**	**TCAM**
500,000	———-	———-	———-
1,000,000	+97.84%	+86.4%	+0%
1,050,000	+107.5%	+89.8%	+0%

Additional hardware resources used when the solution is deployed on top of switch.p4 and as a standalone program

LESSONS LEARNED

- Advantages of using a switch-based relay:
  - Performance

    ~1,000,000 sessions vs ~1,000 sessions per core
  - QoS

    Optimal QoS parameters: delay, delay variation, packet loss rate
  - Flexibility

    The switch permits to modify / forward packets using non-standard fields
  - Timing information

    Measuring delay and its variation on the P4 switch results in precise high-resolution timing information
  - Programmer can free unused resources and customize program

    Accommodate additional sessions
- Limited resources
- Avoid complex application logic

ACKNOWLEDGEMENT

- Thanks to the National Science Foundation (NSF)!

- Activities in the CI Lab at the University of South Carolina are supported by NSF, Office of Advanced Cyberinfrastructure (OAC)

ADDITIONAL SLIDES

DESIGN REQUIREMENTS

- Quality of Service (QoS) parameters
  - Bandwidth
  - Delay
  - Jitter
  - Loss

QoS requirements; stringency of applications[1]

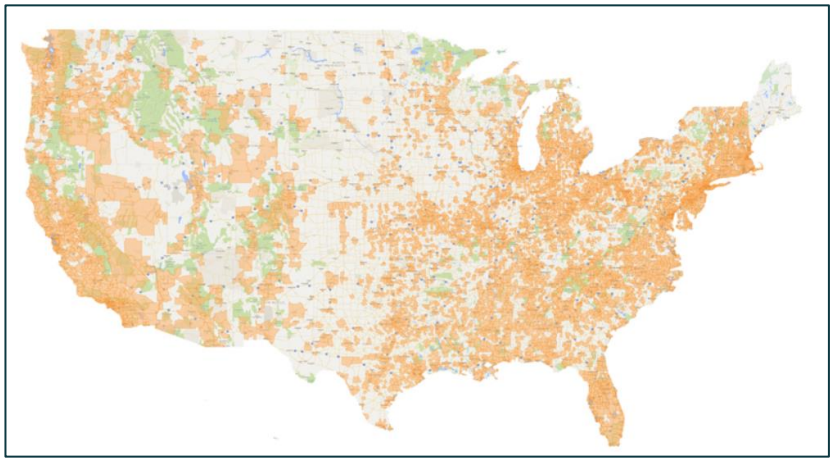
Application	Bandwidth	Delay	Jitter	Packet Loss
VoIP	Low	High	High	Low
Video conference	High	High	High	Low
Data (e.g., file transfer)	High	Low	Low	Medium

[1]A. Tanenbaum, D. Wetherall, “Computer Networks,” Pearson, 5th Edition, p. 405, 2011

MOTIVATION

- According to estimations, media traffic represents approximately 80% of the total traffic over the Internet
  - Much of it is generated by end users communicating with each other
- Media services (voice, video) running alongside the data network in campuses are becoming standard
- Wide Area Networks (WANs) connect centers, campuses
  - E.g., SIP Trunk Network CenturyLink; 10,000 centers, 10,000 centers, 3 billion minutes of voice over IP (VoIP) conversations per month

SIP Trunk Network, CenturyLink



https://tinyurl.com/som38qv